IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

APPLICANTS: Tetsu KIMURA et al.

SERIAL NO.: 10/057,011 GROUP ART UNIT: 2171

FILING DATE: January 25, 2002 EXAMINER: Uyen T. Le

TITLE: RECORDING AND/OR REPRODUCTION APPARATUS, FILE MANAGEMENT METHOD AND PROVIDING MEDIUM

Hon. Commissioner of Patents and Trademarks,
Washington, D.C. 20231

S I R:
CERTIFIED TRANSLATION

I, Masaaki Iwami of 3-22, Asagaya-minami 1-chome, Suginami-ku, Tokyo, Japan, am an experienced translator of the Japanese language into the English language and I hereby certify that the attached comprises an accurate translation into English of Japanese Patent Application No. Hei 09-288178 filed October 21, 1997.

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

December 6, 2004

Date

Masaaki IWAMI

【Document Name】 Application for Patent

【Reference No.】 9800130901

【Application Date】 February 27, 1998

【Destination】 Commissioner, Patent Office

【International Patent Classification】 G11B 7/00

【Title of the Invention】 File System for Recording and/or Reproduction Apparatus Which Uses Disc Type Recording Medium

【Number of Claims】 2

【Inventor】

【Address】 c/o Sony Corporation
7-35, Kitashinagawa 6-chome,
Shinagawa-ku, Tokyo

【Name】 Tetsu Kimura

【Inventor】

【Address】 c/o Sony Corporation
7-35, Kitashinagawa 6-chome,
Shinagawa-ku, Tokyo

【Name】 Akio Ishikawa

【Applicant for Patent】

【Identification No.】 000002185

【Name】 Sony Corporation

【Representative】 Nobuyuki Idei

【Agent】

【Identification No.】 100082131

【Patent Attorney】

【Name】 Yoshio Inamoto

【Telephone No.】 03-3369-6479

【Designation of Charge】

【Ledger No. for Prepayment】 032089

【Amount of Payment】 21,000 yen

【List of Filed Document】

【Object Name】 Specification 1

【Object Name】 Drawing 1

【Object Name】 Abstract 1

【Number of Comprehensive Power of Attorney】 9708842

【Necessity of Confirmation】 Necessary

[NAME OF THE DOCUMENT] Specification

[TITLE OF THE INVENTION] File System for Recording and/or Reproduction Apparatus Which Uses Disc Type Recording Medium

[CLAIMS]

[Claim 1] A file system for a recording and/or reproduction apparatus which uses a disc type recording medium, characterized in that it comprises an AV file system for recording AV data, and management information of said AV file system is recorded in an MIA (Management Information Area) and said MIA is recorded at least at two locations of a logical volume.

[Claim 2] A file system according to claim 1, characterized in that data at least of defect sectors and unused sectors are recorded in said MIA.

[DETAILED DESCRIPTION OF THE INVENTION]

[0001]

[Technical Field to Which the Invention Belongs]

This invention relates to a file system to be used for a recording and/or reproduction apparatus (VDR: Video Disc Recorder) in which a disc type recording medium is used.

[0002]

[Prior Art]

As a file system for recording data onto a disc type recording medium, the ISO/IEC13346, 1995, "Information technology - Volume and file structure of write-once and rewritable media using non-sequential recording for information interchange" is known. This file system is a general purpose file system for recording various data and is not intended to be used by an individual to record compressed digital AV (audio and video) signals onto a disc in a home. Accordingly, the system is not necessarily satisfactory to record compressed digital AV (audio and video) signals. Accordingly, a file system and a volume optimum to record AV signals are demanded.

[0003]

[Subject to Be Solved by the Invention]

Accordingly, a file system with which an individual can record and reproduce AV signals onto and from a disc readily in a home is required.

[0004]

[Means to Solve the Subject]

According to the present invention, a file system for a recording and/or reproduction apparatus which uses a disc type recording medium is characterized in that it comprises an AV file system for recording AV data, and

management information of the AV file system is recorded in an MIA (Management Information Area) and the MIA is recorded at least at two locations of a logical volume.

[0005]

[Embodiment of the Invention]

First, a volume of a file system of the present invention is described in detail.

[0006]

Volume Recognition

2.6.1 Physical Sector

In the present paragraph, the physical sector is defined in the following manner.

● The physical sector number must be an integer which monotonously increases one by one beginning with 0. The physical sector size must be 2^n ($n > 8$).

[0007]

2.6.2 Partition

A storage area defined by successive physical sector numbers is called partition. Further, a number for specifying a partition is called partition number. One physical volume can be divided into a plurality of partitions. The partition number must be an integer which monotonously increases one by one beginning with 0. All of physical sectors in a partition must have an equal

physical size.

Information regarding partitions is defined as a table of partition information in the volume information descriptor. The partition information defines the partition with the physical sector number of a physical sector at the top of the partition and the total number of the physical sectors. One or more partitions must be defined in the physical volume. Where a plurality of partitions are defined, the areas must not overlap with each other.

[0008]

2.6.3 Logical Volume

An aggregate of partitions is called logical volume. The logical volume may be formed from partitions which belong to different physical volumes. The logical sector number must be an integer which monotonously increases one by one beginning with 0. All of physical sectors of partitions which form a logical volume must have an equal physical sector size. The logical sector size must be a multiple of the physical sector size.

Information regarding the logical volume is defined by the volume information descriptor list. The volume information list is a linear single direction list wherein a volume information descriptor is used as a node.

The volume information descriptor list is stored as two main and sub lists in order to assure the reliability.

The volume information descriptor represents a list structure with a next node physical sector number in a volume information descriptor header. The last node of the list is a descriptor whose next node physical sector number is 0 or a descriptor which designates a physical sector whose contents are not suitable as a volume information descriptor. And, a descriptor which stores the latest data is the last node of the list.

The latest data of the sub volume information descriptor list must describe contents same as the latest data of the main volume information descriptor.

The storage locations of the main/sub volume information descriptor lists are described in an anchor descriptor. In the anchor descriptor, the physical sector numbers of the top physical sectors of the areas in which the top nodes of the lists are stored. However, the top descriptor of a node may be, where the magnitude of the descriptor is smaller than PSS32 [bytes], embedded in the anchor descriptor.

Each descriptor must store beginning with the byte position 0 for each physical sector.

[0009]

6.4 Anchor Point

The anchor point is a location into which information necessary to perform an analysis of the volume structure is stored after it is discriminated that the physical volume is described in accordance with the present standards, and is a start point of a volume structure analysis. Whether the physical volume is described in accordance with the present standards or not is dependent on the discrimination that the KIFS descriptor is present in the volume recognition sequence used by the ISO/IEC13346-2/9.1.2.

The anchor descriptor is stored in the anchor point, and is provided at four locations. The anchor point is recorded.

The physical sector numbers are Ch, 20h, LPSN20h, and LPSNCh. Here, the LPSN is the Last physical sector number.

The anchor descriptor is stored in a physical sector of the anchor point beginning with the byte position 0. In the anchor descriptor, the storage locations of the top nodes of the main/sub volume information descriptor lists, the physical sector size and so forth are stored.

[0010]

2.7 Volume Recognition Structure

2.7.1 Standard Identifier of Volume Structure Descriptor

Table 1 is added to the Standard identifier of the volume structure descriptor prescribed in the ISO/IEC13346-2/9.1.2.

[0011]

[Table 1]

Volume Structure Descriptor Interpretation

Identifier	Interpretation
"KIFS1"	According to clause 2.7.2

[0012]

2.7.2 KIFS Descriptor

A KIFS descriptor must be recorded in such a format as illustrated in Table 2. In a volume recognition sequence, the KIFS descriptor must be described following a descriptor defined by the ISO/IEC13346 without fail.

[0013]

[Table 2]

KIFS Descriptor

BP	Length	Name	Contents
0	1	Structure Type	Uints = 0
1	5	Standard Identifier	bytes = "KIFS1"
6	1	Structure Version	Uints = 1
7	1	Reserved	#00 bytes
8	PSS-8	Structure Data	#00 bytes

[0014]

The structure type 0 must be described.

The Standard identifier must describe the character train ¥KIFS01" in accordance with the ISO646.

The structure Version indicates the version of this descriptor. 1 must be described.

The reserved is reserved for future extension. 0 must be described.

The structure data is reserved for future extension. 0 must be described in all bytes.

[0015]

2.8 Volume Data Structure

2.8.1 Anchor Descriptor

The anchor descriptor must be recorded in a format illustrated in Table 3.

[0016]

[Table 3]

Anchor Descriptor

BP	Length	Name	Contents
0	8	Descriptor tag	tag
8	4	Main Volume Information Descriptor List	Unit32
12	4	Sub Volume Information Descriptor List	Unit32
16	4	Physical Sector Size	Unit32
20	2	flags	Unit32 (see Table 9)
22	10	Reserved	#00 bytes
32	PSS_32	Volume Information Descriptor/Reserved	bytes/#00 bytes

[0017]

The descriptor tag indicates that the descriptor is

reserved for future extension. However, where Embedded is designated in the ags, one descriptor is stored here. Where Embedded is not designated, all bytes must be filled with 0.

[0020]

2.8.2 Volume Information Descriptor

The volume information descriptor may be stored over a plurality of physical sectors. The structure of the volume information descriptor is shown in FIG. 1.

The area from the end of the volume information descriptor to the end of the logical sector is reserved for future extension, and all bytes must be filled with 0.

[0021]

2.8.3 Volume Information Descriptor Header

The volume information descriptor header must be described in accordance with Table 5.

[0022]

[Table 5]

Volume Information Descriptor Header

BP	Length	Name	Contents
0	8	Descriptor tag	tag
8	2	Information Size	Unit16
10	2	Character Set	Chara Set(see table1.5.11)
12	20	Volume Indentifier	bytes
32	256	Volume Name	bytes
288	10	Create Time	Time Stamp(see Table 1.5.10)
295	10	Modify Time	Time Stamp(see Table 1.5.10)
308	2	Number of Partitions	Unit16
310	2	Number of Logical Volume	Unit16
312	2	Reserved Area Identifier	Unit16
314	62	Reserved Area	bytes
376	4	Next Reserved Area Physical Sector Number	Unit32
380	4	Next Node Physical Sector Number	Unit32

[0023]

The Descriptor tag

indicates that this descriptor is a volume information descriptor.

The Information Size

indicates the number of the bytes of this descriptor.

The Character Set

indicates a character code of a physical volume name described in the volume Name field.

The Volume Identifier

indicates a byte train which uniquely defines a physical volume.

The Volume Name

stores a physical volume name.

The Create Time

stores the date and the time when this physical volume is used as the KIFS for the first time.

The Modify Time

stores the date and the time when the latest volume information descriptor of this physical volume is modified.

The Number of Partitions

is the number of partitions included in this physical volume. This must coincide with the number of the partition information.

The Number of Logical Volume

is the number of logical volumes to which the partitions included in this physical volume belong. This must coincide with the number of logical volume information.

The Reserved Area Identifier

is an ID for specifying the standards which use the reserved area.

The Reserved Area

is reserved for further extension. Where this is not used, all bytes must be filled with 0.

The Next Reserved Area Physical Sector Number

stores the physical sector number of the top physical sector of the extended area. In this physical sector, the

reserved area header (See 2.8.8) and extension information are stored successively. Where there is no extended area, 0 is stored.

The Next Node Physical Sector Number stores the physical sector number of the top physical sector in the area in which the next node is stored. Where there is no next node, 0 must be stored.

[0024]

2.8.4 Partition Information

The partition information must be recorded in a format illustrated in Table 6.

[0025]

[Table 6]

Partition Information

BP	Length	Name	Contents
0	4	Start Physical Sector Number	Unit32
4	4	Number of Physical Sectors	Unit32
8	2	Reserved Area Identifier	Unit16
10	62	Reserved Area	bytes
72	4	Next Reserved Area Physical Sector Number	Unit32

[0026]

The start physical sector number stores the physical sector number of the top physical sector of the partition area.

The number of physical sectors stores the total number of physical sectors which belong to the partition area.

The reserved area identifier is an ID for specifying the format which uses the reserved area.

The reserved area is reserved for future extension. Where it is not used, all bytes must be filled with 0.

The next reserved area physical sector number stores the physical sector number of the top physical sector of the extended area. In this physical sector, the reserved area header and the extension information are stored successively. Where there is no extension area, 0 is stored.

[0027]

2.8.5 Logical Volume Information

The structure of the logical volume information is illustrated in FIG. 2.

[0028]

2.8.6 Logical Volume Information Header

The logical volume information header must be recorded in a format illustrated in Table 7.

[0029]

[Table 7]

Local Volume Information Header

Bit	Length	Name	Contents
0	4	Logical Volume Contents Identifier	Uint32
4	2	Character Set	Chara set (see table 1.5.11)
6	2	Number of Partitions	Uint16
8	20	Logical Volume Identifier	bytes
28	256	Logical Volume Name	bytes
284	16	Logical Volume Contents Use	bytes
300	2	Logical Sector Size	Uint16
302	2	Reserved Area Identifier	Uint16
304	62	Reserved Area	bytes
366	4	Next Reserved Area Physical Sector Number	Uint32

[0030]

The logical volume contents identifier specifies data stored in this logical volume.

An ID defined is indicated in Table 8.

[0031]

[Table 8]

Logical Volume Contents Identifier

Identifier	Interpretation
"KIFS1"	According to clause 3

[0032]

The Character Set stores character codes of the logical volume name stored in the logical volume Name field.

The number of partitions stores the number of partitions which form this logical volume. This must coincide with the number of the entries of the partition map.

The logical volume identifier is an ID for uniquely identifying the logical volume.

The logical volume name

The logical volume name

The logical volume contents use

are areas which may be used freely by data stored in this logical volume.

The logical sector size

stores the size (the number of bytes) of the logical sector of this logical volume.

The reserved area identifier

is an ID for specifying the format which uses the reserved area.

The reserved area

is reserved for future extension. Where this is not used, all bytes must be filled with 0.

The next reserved area physical sector number

stores the physical sector number of the top physical sector of the extended area. In this physical sector, the reserved area header and the extension information are stored successively. Where there is no extended area, 0 is stored.

[0033]

2.8.7 Partition Map

The partition map must be recorded in a format illustrated in Table 9.

[0034]

[Table 9]

Partition Map

RBP	Length	Name	Contents
0	20	Volume Identifier	bytes
20	2	Partition Number	Unit16
22	2	Reserved	#00 bytes

[0035]

The volume identifier

is an ID for specifying the physical volume in which the partition is stored.

The partition number

indicates what numbered partition is in the physical volume. The order of description of partition information is an order of the partition.

The reserved

is reserved for future extension. All bytes must be filled with 0.

[0036]

2.8.8. Reserved Area Header

The extended area must be described beginning with the byte position 0 of the physical sector. At the top of the extended area, the header indicated in Table 10 must

be described. The extended area may extend over a plurality of physical sectors. Extended information is stored in an area following the header.

The reserved area identifier

is an ID for specifying the format which uses the reserved area.

The reserved area

is reserved for future extension. Where this area is not used, all bytes must be filled with 0.

The area size

stores the size (the number of physical sectors) of the extended area.

[0037]

[Table 10]

Reserved Area Header

BP	Length	Name	Contents
0	2	Reserved Area Identifier	Unit16
2	2	Reserved	#00 bytes
4	4	Area Size	Unit32

[0038]

2.10 Appendix

2.10.1 Use of the Volume Structure in a Write Media

● Volume Recognition Sequence

This cannot be moved because this is an area whose storage starting position is fixed. The last end of the

sequence is ended not with a Terminating extended area descriptor but with a non-recorded sector, and it is possible to increase the standards to be recognized by adding a volume structure descriptor. However, it is impossible to decrease the standards. It is further preferable to finalize the last end with a Terminating extended area descriptor.

- Anchor Descriptor

This cannot be moved because the storage position thereof is fixed. Once this is written, it is not rewritten. The position of the volume information sequence does not move.

- Volume Information Descriptor List

Since the storage position of the top node of the list cannot be changed because the anchor descriptor cannot be rewritten, it is possible to change the node in which the latest data is stored with the list. More particularly, the physical sector number of an unrecorded physical sector may be stored in the next node physical sector number field so that the physical sector number may be used upon updating.

[0039]

2.10.2 How to Find Out an Empty Area (Physical Sectors)

A remaining area when the following areas (physical

sectors) are excepted from the entire area is an area which is not used by the KIFS.

- Area following 10h in which the volume recognition sequence is described.
- Ch, 20h, LPSN20h, LPSNCh in which an anchor descriptor is stored.
- Area in which the nodes of the volume information descriptor list are stored
- Physical sector indicated, when the next node physical sector number of the node in which the latest data of the volume information descriptor list is stored is not 0, by the field.
- Partition area designated by the partition descriptor.
- Extended area indicated by the next reserved area d physical sector number fields of the volume information descriptor header, partition information and logical volume information header.

[0040]

182. Volume Structure

2.10.3 Volume Recognition Sequence

The flow of recognition of a volume structure is such as described below.

1. The volume recognition sequence stored in the area beginning with PSN10h is interpreted based on the

ISO/IEC13346-2/8 to recognize that this physical volume is described in accordance with the format of the KIFS.

2. The anchor descriptor written in PSNCh, 20h, LPSN-20h, LPSN-Ch is read to recognize the storage locations of the logical sector size and the volume information list.

3. The volume structure is recognized using the latest data in the volume information list.

[0041]

2.10.4 Example of the Volume Structure

Table 11 illustrates an example of the volume structure of a hybrid disc of the FAT, ISO9660 (MSJoliet), ISO/IEC13346, KIFS. ◆ represents position fixed information which cannot be re-arranged.

[0042]

[Table 11]

Example of volume structure (HFS,9660,13346,KIFS Hybrid)

PSN(hax)	Descriptor	Contents
0	[FAT]Partition Table	◆[FAT]Partition Table
-	-	-
c	[KIFS]Anchor Descriptor	◆[KIFS]Anchor
d	[KIFS]Volume Information Descriptor	[KIFS]Main Volume Information Sequence
e	[KIFS]Termination Descriptor	
f	-	-
10	[9660]Primary Volume Descriptor	◆[9660/13346/KIFS] Volume recognition Sequence
11	[9660]Primary Volume Descriptor(sub)	
12	[9660]Supplementary Volume Descriptor(for Joliet)	
13	[9660]Volume Descriptor Set Terminator	
14	[13346]Beginning Extended Area Descriptor	
15	[13346]NSR Descriptor	
16	[KIFS]KIFS Descriptor	
17	[13346]Terminating Extended Area Descriptor	
-	-	-
20	[KIFS]Anchor Descriptor	◆[KIFS]Anchor
21	[KIFS]Volume Information Descriptor	[KIFS]Sub Volume Information Sequence
22	[KIFS]Termination Descriptor	
-	-	-
30	[13346]Primary Volume Descriptor	[13346]Main Volume Descriptor Sequence Extent
31	[13346]Implementation Use Volume descriptor	
32	[13346]Partition Descriptor	
33	[13346]Logical Volume Descriptor	
34	[13346]Unallocated Space Descriptor	
35	[13346]Terminating Descriptor	
-	-	-
40	[13346]Primary Volume Descriptor	[13346]Reserve Volume Descriptor Sequence Extent
41	[13346]Implementation Use Volume Descriptor	
42	[13346]Partition Descriptor	
43	[13346]Logical Volume Descriptor	
44	[13346]Unallocated Space Descriptor	
45	[13346]Terminating Descriptor	
-	-	-
100	[13346]Anchor Volume Descriptor Pointer	◆[13346]Anchor
	-	-
150	[KIFS]LOGICAL VOLUME	
-	-	
LPSN-150	-	
-	-	-
LPSN-100	[13346]Anchor Volume Descriptor Pointer	◆[13346]Anchor
-	-	-
LPSN-20	[KIFS]Anchor Descriptor	◆[KIFS]Anchor
-	-	-
LPSN-c	[KIFS]Anchor Descriptor	◆[KIFS]Anchor
-	-	-
LPSN	[13346]Anchor Volume Descriptor Pointer	◆[13346]Anchor

[0043]

3 AV File System

3.3.1 Logical Sector

An area into and from which data of a size of 2^n (n>8) bytes can be reproduced or recorded/reproduced.

[0044]

The logical sector number is a number applied for identification of the logical sector.

[0045]

3.3.3 Logical Volume

A set composed of logical sectors which have consecutive ascending logical sector numbers beginning with 0 and have an equal size.

[0046]

3.3.4 Management Information Area (MIA)

An area composed of a plurality of successive logical sectors on the logical volume for storing various kinds of control information of the AV file system.

[0047]

3.3.5 Management Information Block

A logical sector in the MIA.

[0048]

3.3.6 Management Information Block Number

A value obtained by subtracting the logical sector number of the top of the management information block of the MIA from the logical sector number of the management information block.

[0049]

3.3.7 Allocation Range Chain

An area on the logical volume composed of one or more allocation extents.

[0050]

3.6 AV File System Overview

3.6.1 AV File System Descriptor

The AV file system descriptor is recorded in one logical sector on the logical volume and designates the positions and the sizes of a main MIA and a reserved MIA on the logical volume and the positions of the MIA maps on the MIAs.

Where the system which defines the logical volume has a capability of describing attribute information of the logical volume, the position of the AV file system descriptor is designated using this function. Where the system which defines the logical volume does not have such a function, at least one AV file system descriptor must be placed at the top of the logical volume.

In order to assure the reliability, a plurality of

AV file system descriptors can be recorded in the logical volume.

For the position of the AV file system descriptor where the logical volume is prescribed by the clause 2, it is planned to use the logical volume contents Use field of the TBD logical volume information header. [0051]

3.6.2 Management Information Area (MIA)

Various kinds of management information of the AV file system are recorded in the management information area (MIA). In order to assure the reliability, an MIA having management information of the same contents is recorded at two locations on the logical volume, and they are individually called main MIA and reserved MIA. The positions and the sizes of the main MIA and the reserved MIA and the positions of the MIA maps in the MIAs are defined by the AV file system descriptor.

A logical sector in an MIA is called management information block, and the offset of the logical sector number from the top sector of the MIA is called management information block number. For designation of a sector in an MIA, the management information block number is used.

The MIA is composed of sectors which cannot be used

because of defects or the like, unused sectors, and sectors used to store the following data structures:

- an MIA map,
- a file table,
- an allocation extents table,
- an allocation structure table,
- a defect information table (Optional), and
- an extended attribute table (Optional).

For what objects the sectors in the MIA are used is recorded in the MIA map.

Each of the various data structures may be stored in one management information block or over a plurality of management information blocks. Where a data structure is recorded over a plurality of management information blocks, which management information blocks should be linked in what order is recorded in the map Entries field in the MIA map.

If a data structure ends intermediately of a management information block, then the bytes beginning with the byte next to the end of the data and ending with the last byte of the management information block must be filled with #00.

[0052]

3.6.3 File Table

In the AV file system, files and directories are managed with a file table. The file table is composed of a file table header and a file record, and information defining the structure and the number of file records is stored in the file table header while information regarding the individual files and directories is stored in the file records. The structure of the file records is designated by the file system structure type field of the file table header, and only the type 0 is prescribed at present.

In the file system structure type 0, a file record is a data area of a fixed length and is composed of

- a field for identifying the file record,
- a field which represents the type of the file record,
- a field which represents the dates and the times of production and modification,
- a field which represents the position and the size of the data
- a field which represents an attribute,
- a field which indicates a parent file record called parent link,
- a field which indicates a brother file record called next link,
- a field which indicates a child file record called

child link, and

- a field which indicates an extended attribute record chain.

The file records are numbered with consecutive ascending numbers beginning with 0, and the numbers are called file record numbers. The parent link, next link and child link are designated using a file record number.

In the file system structure type 0, such a hierarchical structure as shown in FIG. 3 can be constructed using the first file record in the file table as a Root. Each circle in FIG. 3 represents one file record, and the top file record corresponds to the Root.

[0053]

This hierarchical structure is implemented by setting the parent link, next link and child link in such a manner as shown in FIG. 4.

[0054]

The numeral of the Figure in the text is not correct from the problem of the style file of the TeX. Further, in the parent link in the bottom figure, an arrow mark by which the Root file record indicates itself is not written.

A list of file records formed from next links is called file record chain, and the list must not include

two or more records which have the same file ID and besides have the same file type.

Where file records are constructed such that nodes other than leaf nodes do not have data, an index node corresponds to a directory of a popular file system, and a leaf node corresponds to a file.

[0055]

3.6.6 Defect Information Table

The defect information table is a table in which logical sector numbers of defect sectors in the logical volume are recorded, and can be used for management of defect sectors. Mounting of this table is Optional.

[0056]

3.6.7 Extended Attribute Table

The extended attribute table can be used to hold an extended attribute of a file or a directory in the MIAs. Mounting of this table is Optional. It is presumed to use the extended attribute table when to apply a file name.

[0057]

3.7 Signature

The Signature is set at the top of the data structure used by the AV file system. The Signature must be recorded in such a manner as illustrated in Table 12.

[0058]

[Table 12]

Signature

RBP	Length	Name	Contents
0	4	Identification	bytes = "AVFS"
4	1	Version	Unit8 = 1
5	1	Data type	Units
6	2	Reserved	#00 bytes

[0059]

In the Identification (RBP 0), the character train "AVFS" must be set in accordance with the ISO646.

The Version (RBP 4) designates the version number of the present standards. Where the present standards are used, it must be set to 1.

The data type (RBP 5) designates the type of the data structure. One of values illustrated in Table 13 must be set depending upon the type of the data structure.

[0060]

[Table 13]

Data type

Value	Interpretation
0	Reserved
1	AV File System Descriptor
2	MIA Map
3	File Table
4	Allocation Extents Table
5	Allocation Strategy Table
6	Defect Information Table
7	Extended Attribute Table
8-255	Reserved

[0061]

The reserved (RBP 6) is reserved for extension and must have #00 set therein.

The Signature is presumed to be used to identify the data structure upon crash recovery.

[0062]

3.8 AV File System Descriptor

The AV file system descriptor must be recorded in such a manner as shown in Table 14.

[0063]

[Table 14]

AV File System Descriptor

BP	Length	Name	Contents
0	8	Signature	Signature
8	4	Location of Main MIA	Unit32
12	4	Location of Reserve MIA	Unit32
16	2	Length of Main MIA	Unit16
18	2	Length of Reserve MIA	Unit16
20	4	Creation time	Unit32
24	4	Modification Time	Unit32
28		Number of MIA Map Sectors in Main MIA ($= x_1$)	Unit16
30	2	Number of MIA Map Sectors in Reserve MIA ($= x_2$)	Unit16
32	$2x_1$	MIA Map sectors in Main MIA	bytes
$32+2x_1$	$2x_2$	MIA Map sectors in Reserve MIA	bytes

[0064]

The Signature (BP 0) is defined in 3.7. The data type field of the Signature must have 1 set therein.

The location of main MIA (BP 8) designates the sector number of the start logical sector of the main MIA.

The location of reserve MIA (BP 12) designates the

sector number of the start logical sector of the reserve MIA.

The length of main MIA (BP 16) designates the size of the main MIA in the form of the number of logical sectors.

The length of reserve MIA (BP 18) designates the size of the reserve MIA in the form of the number of logical sectors.

The Creation Time (BP 20) designates the date and the time when the AV file system descriptor is produced.

The Modification Time (BP 24) designates the date and the time when the AV file system descriptor is updated.

The number of MIA map sectors in main MIA (BP 28) designates the number of MIA sectors described in the main MIA map sectors (BP 32).

The number of MIA map sectors in reserve MIA (BP 30) designates the number of reserve MIA sectors described in the reserve MIA map sectors (BP $32+2x_1$).

The MIA map sectors in main MIA (BP 32) designates management information blocks which form the MIA map in the main MIA. The management information block numbers of the management information blocks which form the MIA map must be set in order.

The MIA map sectors in reserve MIA (BP $32+2x_1$) designates management information blocks which form the MIA map in the reserve MIA. The management information block numbers of the management information blocks which form the MIA map must be set in order.

[0065]

3.9 MIA Map

The MIA map is used to indicate a situation of use of management information blocks in the MIAs. The MIA Map indicates the locations of various data structures in the MIAs, defect sectors, and unused sectors.

The MIA map must be recorded in such a manner as seen in Table 15.

[0066]

[Table 15]

MIA Map

BP	Length	Name	Contents
0	8	Signature	Signature
8	2	Location of MIA Map	Unit16
10	2	Location of Allocation Strategy Table	Unit16
12	2	Location of File Table	Unit16
14	2	Location of Allocation Extents Table	Unit16
16	2	Location of Defect List Table	Unit16
18	2	Location of Extended Attribute Descriptor	Unit16
20	2	Reserved	bytes
22	2	Number of Map Entries (= x_1)	Unit16
24	$2x_1$	Map Entries	bytes

[0067]

The Signature (BP 0) is defined in 3.7. The data type field of the Signature must have 2 set therein.

The location of MIA map (BP 8) designates the management information block number of the top sector of the MIA map in this MIA.

The location of allocation strategy table (BP 10) designates the management information block number of the top sector of the allocation strategy table in this MIA.

The location of file table (BP 12) designates the management information block number of the top sector of the file table in this MIA.

The location of allocation extents table (BP 14) designates the management information block number of the top sector of the allocation extent table in this MIA.

The location of defect list table (BP 16) designates the management information block number of the top sector of the defect list table in the MIA. If the MIA does not have a defect list table therein, then #FFFF must be set.

The location of the extended attribute descriptor (BP 18) designates the management information block number of the top sector of the extended attribute descriptor in this MIA. If this MIA does not have an extended attribute descriptor therein, then #FFFF must be set.

The reserved (BP 20) is reserved for extension and

must have #00 set therein.

The number of map Entries (BP 22) designates the number of entries of map Entries beginning with (BP 24). This number must be equal to the number of logical sectors existing in the MIA.

The map Entries (BP 24) designate a situation of use of the logical sectors in this MIA. One map Entry is composed of Uint16, and the first map Entry corresponds to the first management information block of the MIA, the second map Entry corresponds to the second management information block, …….

The value of the map Entries has such meanings as indicated in Table 16.

[0068]

[Table 16]

Map entry value

Value	Interpretation
#0000-#FFEF	Next Management Information Block Number
#FFF0	Defective sector
#FFF1	Unused Management Information Block
#FFF2-#FFFE	Reserved
#FFFF	Last Management Information Block of the data structure

[0069]

If the data structure has a size equal to or smaller than the size of a management information block and is stored in one management information block, then #FFFF must be placed in the map Entry corresponding to

the management information block. Where the data structure is recorded over a plurality of management information blocks, in the map Entry corresponding to each of the management information blocks other than the last management information block, the management information block number of the next management information block must be placed. Any management information block whose map Entry has the value #FFF1 indicates that the block is unused and can be used where the data structure requires a new management information block. Any management information block whose map Entry has the value #FFF0 indicates that the block is defective (because of a defective sector or the like).

[0070]

3.10 File Table

3.10.1 Structure of the File Table

Information regarding files and directories is recorded in a file record in the file table. The file table is composed of a file table header and a file record as described below.

[Allocation extents table]{

<File table header>1

<File record>1+n

}

[0071]

3.10.2 File Table Header

The file table header must be recorded in such a manner as seen from Table 17.

[0072]

[Table 17]

File Table Header

BP	Length	Name	Contents
0	8	Signature	Signature
8	2	Number of File Records	Unit16
10	2	File System Structure Type	Unit16
12	4	Reserved	#00 bytes

[0073]

The Signature (BP 0) is defined in 3.7. The data type field of the Signature must have 3 set therein.

The number of file records (BP 8) designates the number of the file records in the file table.

The file system structure type (BP 10) defines the structure of the file records and the relationship between the file records. The value indicated by this field signifies the meaning illustrated in Table 18.

[0074]

[Table 18]

File System Structure Type

Value	Interpretation
0	File System Structure Type 0 (see 3.10.4)
1-65535	Reserved

[0075]

The reserved (BP 12) is reserved for extension and must have #00 set therein.

[0076]

3.10.3 File Record

The structure of file records depends fully on the file system structure type field of the file table header, and an item which is common to all file system structure types does not exist.

[0077]

3.10.4 File Record of the File System Structure Type 0

With the file table structure type 0, the file record must be recorded in such a manner as indicated in Table 19.

[0078]

[Table 19]

File Record of File Structure Type 0

RBP	Length	Name	Contents
0	2	File ID	Unit16
2	2	File Type	Unit16
4	4	Attribute	Unit32
8	4	Creation Time	Unit32
12	4	Modification Time	Unit32
16	8	Data Length	Unit64
24	4	Data Location	Unit32
28	2	Allocation Strategy Number	Unit16
30	2	Child Link	Unit16
32	2	Next Link	Unit16
34	2	Parent Link	Unit16
36	2	Extended Attribute Record	Unit16
38		Number	#00 bytes

[0079]

The file ID (RBP 0) designates a number for identifying a file record having a same file type in the file record chain.

The file type (RBP 2) designates a number for indicating the type of this file record (See 3.10.5).

The attribute (RBP 4) designates the attribute of the file record or data to be referred to by the file record (See 3.10.6).

The Creation Time (RBP 8) designates the date and the time of production of the file record.

The Modification Time (RBP 12) designates the date and the time of modification to the file record or to the data to be referred to by the file record.

The data length (RBP 16) designates the length of the data to be referred to by the data location (RBP 24) in units of a byte.

The data location (RBP 24) designates the allocation extents record number to be referred to by the file record.

The allocation structure number (RBP 28) designates in accordance with which allocation structure the data designated by the data location (RBP24) is allocated.

The child link (RBP 30) designates the file record

number of the child file record. If no such file record is present, #FFFF must be placed.

The next link (RBP 32) designates the file record number of a next file record which composes the file record chain. If no such file record is present, then #FFFF must be placed.

The parent link (RBP 34) designates the file record number of the parent file record. If no such file record is present, then #FFFF must be placed.

The extended attribute record number (RBP 36) designates the top extended attribute record number of the extended attribute record chain used by this file record. If the extended attribute record is not referred to, then #FFFF must be placed.

The reserved (RBP 38) is reserved for extension and must have #00 set therein.

[0080]

3.11 Allocation Extents Table

3.11.1 Structure of the Allocation Extents Table

Information regarding the allocation extents is recorded on the allocation extent records of the allocation extents table which correspond in a one-by-one corresponding relationship. The allocation extents table is composed of an allocation extents table header and an

allocation extent record as indicated below.

[Allocation Extents Table] {

<Allocation Extents Table Header>1

<Allocation Extent Record>n

}

[0081]

Whether an allocation extent record has an allocation extent corresponding thereto is designated in a field of the attribute of the allocation extent record. Allocation extent records which are not used in the allocation extents table form one list using the next allocation extent record field, and this is called Free allocation extent record chain.

[0082]

3.11.2 Allocation Extents Table Header

[Table 20]

Allocation Extents Table Header

BP	Length	Name	Contents
0	8	Signature	Signature
8	2	Number of Allocation Extent Records	Unit16
10	2	First Free Allocation Extent Record	Uint16
12	4	Reserved	#00 bytes

[0083]

The Signature (BP 0) is defined in 3.7. The data type field of the Signature must have 4 set therein.

The number of allocation extent records (BP 8)

designates the number of allocation extent records in the allocation extents table.

The first free allocation extent record (BP 10) designates the first element of the Free allocation extent record chain. If the allocation extents table has no free allocation extent record, then #FFFF must be placed.

The reserved (BP 12) is reserved for extension and must have #00 set therein.

[0084]

3.11.3 Allocation Extent Record

The allocation extent record represents the start position, the end position and the attribute of the Allocation extent and the position of the next allocation extent record which forms the allocation extent chain. The allocation extent record is recorded in such a manner as indicated in Table 21.

[0085]

[Table 21]

Allocation Extent Record

RBP	Length	Name	Contents
0	4	Start Logical Sector Number	Unit32
4	1	Allocation Strategy Number	Unit8
5	1	Reserved	Unit8
6	2	Start Offset	Unit16
8	4	End Logical Sector Number	Unit32
12	2	Reserved	Unit16
14	2	End Offset	Unit16
16	4	Attribute	Unit32
20	4	Next Allocation Extent Record	Unit32
24	8	Length of the Allocation Extent	Unit64

[0086]

The Start logical sector number (RBP 0) designates a logical sector number of a logical sector which includes the start byte of the allocation extent.

The allocation strategy number (RBP 4) indicates in accordance with which allocation strategy the allocation extent record is arranged.

The reserved (RBP 5) is reserved for extension and must have #00 set therein.

The Start Offset (RBP 6) designates a byte offset from the top byte of the logical sector including the start byte of the allocation extent to the start byte. The Start Offset has 0 placed therein if the start position is equal to the top byte of the logical sector.

The End logical sector number (RBP 8) designates

this field.

[0090]

3.12 Allocation Strategy Table

3.12.1 Allocation Strategy Table Structure

The allocation strategy table is used to designate all allocation strategies used to arrange data in this logical volume by the AV file system. The allocation strategy table is composed of an allocation strategy table header and an allocation strategy record as seen below.

```
[Allocation Strategy Table] {
    <Allocation Strategy Table Header>1
    <Allocation Strategy Record>n
}
```

[0091]

3.12.2 Allocation Strategy Table Header

The allocation strategy table header must be recorded in such a manner as seen in Table 23.

[0092]

[Table 23]

Allocation Strategy Table Header

BP	Length	Name	Contents
0	8	Signature	Signature
8	2	Number of Allocation Strategy Record	Unit16
10	6	Reserved	#00 bytes

[0093]

The Signature (BP 0) is defined in 3.7. The data type field of the Signature must have 5 set therein.

The number of allocation strategy record (BP 8) designates the number of allocation strategy records in the allocation strategy table.

The reserved (RBP 10) is reserved for extension and must have #00 placed therein.

[0094]

12.3 Generic Format of the Allocation Strategy Record

An allocation strategy record is used to designate an allocation strategy. The allocation strategy record must be recorded in such a manner as seen in Table 24.

[0095]

[Table 24]

Allocation Strategy Record

RBP	Length	Name	Contents
0	2	Length of Allocation Strategy Record	Unit16
2	2	Allocation Strategy Type	Uint16
4	1	Allocation Strategy Number	Unit8
5	3	Reserved	#00 bytes
8	x_1	Allocation Strategy Type Dependent Date	bytes

[0096]

The length of allocation strategy record (RBP 0) designates the length of this allocation strategy record in the form of the number of bytes. The length of the

allocation strategy record must be a multiple of 8.

The allocation strategy type (RBP 2) designates the type of this allocation strategy record. (See 3.12.4, 3.12.5)

The allocation strategy number (RBP 4) designates what numbered record in the allocation strategy table this allocation strategy record is. If this record is the first record, then 0 is placed.

The reserved (RBP 5) is reserved for extension and must have #00 placed therein.

The allocation strategy type dependent data (RBP 8) has set therein contents which are determined for each allocation strategy type.

[0097]

3.12.4 Allocation Strategy Type 0

In the allocation strategy type 0, the following conditions:

1. that the allocation extent must be arranged in an area designated by the Start logical sector number (RBP 8) and the End logical sector number (RBP 12) of the allocation strategy record;

2. that, where part of the logical sector is allocated to a certain allocation extent, any byte of the logical sector must not belong to another allocation

extent; and

3. that the top of the allocation extent and the top of the logical sector must coincide with each other; must be satisfied.

[0098]

The allocation strategy record of the allocation strategy type 0 must be recorded in such a manner as seen in Table 25.

[0099]

[Table 25]

Allocation Strategy Record of Allocation Strategy Type 0

RBP	Length	Name	Contents
0	2	Length of Allocation Strategy Record	Unit16 (=16)
2	2	Allocation Strategy Type	Unit16 (= 0)
4	1	Allocation Strategy Number	Unit8
5	3	Reserved	#00 bytes
8	4	Start Logical Sector Number	Unit3
12	4	End Logical Sector Number	Unit32

[0100]

The length of allocation strategy record (RBP 0) must have 16 set therein.

The allocation strategy type (RBP 2) must have 0 set therein.

The allocation strategy number (RBP 4) designates what numbered record the allocation strategy record is in the allocation strategy table. If this record is the first record, then 0 is set.

The reserved (RBP 5) is reserved for extension and must have #00 set therein.

The Start logical sector number (RBP 8) designates the top logical sector number in which the allocation extent is arranged.

The End logical sector number (RBP 12) designates the last logical sector number of the area in which the allocation extent is arranged.

[0101]

3.12.5 Allocation Strategy type 1

The allocation strategy record of the allocation strategy type 1 must be recorded in such a manner as seen in Table 26.

[0102]

[Table 26]

Allocation Strategy Record of Allocation Strategy Type 1

RBP	Length	Name	Contents
0	2	Length of Allocation Strategy Record	Unit16
2	2	Allocation Strategy Type	Unit16 (= 1)
4	1	Allocation Strategy Number	Unit8
5	3	Reserved	#00 bytes
8	2	Number of Zones (= x_1)	Unit16
10	6	Reserved	#00 bytes
16	$16x_1$	Zone Information Records	(see Table 32)

[0103]

The length of allocation strategy record (RBP 0) must have the length of the allocation strategy record, that is, $16+16x_1$, set therein.

The allocation strategy type (RBP 2) must have 1 set therein.

The allocation strategy number (RBP 4) designates what numbered record this allocation strategy record is in the allocation strategy table. If this record is the first record, then #00 must be set.

The reserved (RBP 5) is reserved for extension and must have #00 set therein.

The number of zones (RBP 8) designates the number of zone information records in the allocation strategy record.

The reserved (RBP 10) is reserved for extension and must have #00 set therein.

The zone information records (RBP 16) must have placed therein a number of zone information records which is designated by the number of zones (RBP 8). The zone information records must be recorded in such a manner as seen in Table 27.

[0104]

[Table 27]

Zone Information Record

RBP	Length	Name	Contents
0	4	Start Logical Sector Number	Unit32
4	4	End Logical Sector Number	Unit32
8	4	Length of Allocation Unit	Unit32
12	4	Reserved	#00 bytes

[0105]

The Start logical sector number (RBP 0) designates the start logical sector number of this zone.

The End logical sector number (RBP 4) designates the last logical sector number of this zone.

The length of allocation Unit (RBP 8) designates an allocation Unit to be arranged into this zone.

The reserved (RBP 12) is reserved for extension and must have #00 placed therein.

[0106]

3.13 Defect Informatin Table

The defect information table is used to record logical sector numbers of defect sectors in the logical volume. The defect information table must be recorded in such a manner as seen in Table 28.

[0107]

[Table 28]

Defect Information Table

BP	Length	Name	Contents
0	8	Signature	Signature
8	4	Number of Defect Sectors (= x_1)	Unit32
12	4	Reserved	#00 bytes
16	$4x_1$	Defect Sector Addresses	bytes

[0108]

The Signature (BP 0) is defined in 3.7. The data type field of the Signature must have 6 set therein.

The number of detect sectors (BP 8) designates the number of entries of the defect sector Addresses beginning with (BP 16).

The reserved (BP 12) is reserved for extension and must have #00 set therein.

The defect sector addresses (BP 16) designate logical sector numbers of defect sectors detected in the logical volume. One entry is composed of Unit32, and the values recorded here must be sorted in an ascending order. [0109]

3.14 Extended Attribute Table

3.14.1 Extended Attribute Table Structure

The extended attribute table is composed of an extended attribute table header and an extended attribute record as indicated below.

```
[Extended Attribute Table] {
    <Extended Attribute Table Header>1
    <Extended Attribute Record>n
}
```

[0110]

The extended attribute records in the extended attribute table are numbered with extended attribute record numbers consecutively in an ascending order

beginning with 0, and a list can be produced by placing this number into the next extended attribute record field of the extended attribute record. This list is called extended attribute record chain.

Those extended attribute records which are not used in the extended attribute table are managed in a list called Free extended attribute record chain.

[0111]

3.14.2 Extended Attribute Table Header

The extended attribute table header must be recorded in such a manner as seen in Table 29.

[0112]

[Table 29]

Extended Attribute Table Header

BP	Length	Name	Contents
0	8	Signature	Signature
8	2	Number of Extended Attribute Record	Unit16
10	2	First Free Extended Attribute Record	Unit16
12	4	Reserved	#00 bytes

[0113]

The Signature (BP 0) is defined in 3.7. The data type field of the Signature must have 7 set therein.

The number of extended attribute record (BP 8) designates the number of extended attribute records in the extended attribute table.

The First Free extended attribute record (BP 10)

designates the first element of the Free extended attribute record chain. If the extended attribute table does not include a free extended attribute record therein, then #FFFF must be set.

The reserved (BP 12) is reserved for extension and must have #00 set therein.

[0114]

3.14.3 Extended Attribute Record

The extended attribute record must be recorded in such a manner as seen in Table 30.

[0115]

[Table 30]

Extended Attribute Record

RBP	Length	Name	Contents
0	2	Next Extended Attribute Record	Unit16
2	30	Extended Attribute Information	bytes

[0116]

The next extended attribute record (RBP 0) designates the next extended attribute record number which composes the extended attribute record chain. If the extended attribute record is the last extended attribute record, then #FFFF must be set.

[0117]

5.1 Management of Defect Sectors

5.1.2 Volume Structure

The definition of the volume structure is given by

- volume recognition sequence,
- anchor descriptor, and
- volume information descriptor.

Countermeasures to defect sectors for those kinds of information are taken in the following manner.

- Volume recognition sequence

This definition is given in the ISO/IEC13346 and cannot be specified in the present standards. Since the area into which this sequence is to be stored is fixed, if a defect sector exists there and replacement processing is not performed by a drive, then the sequence cannot be recorded correctly.

However, the volume recognition sequence merely indicates based on what standards the physical volume is used, and if this information is not read normally, it can be discriminated from presence or absence of the anchor descriptor.

- Anchor descriptor

This is recorded in a fixed sector so that the location may be specified even if the volume recognition cannot be read.

This is recorded at four locations in an

overlapping relationship taking a case wherein the record sector is defective into consideration.

- Volume information descriptor list

This can be recorded in a free region by writing the recording location of the top area of this list in the anchor descriptor. This can be recorded in a non-defective sector by performing write-and-verify without fail upon recording. Further, taking a case wherein a defect occurs after recording into consideration, this is recorded at two locations in an overlapping relationship.

[0118]

5.1.3 AV file system

As topics:

- AV file system descriptor
- MIA reliability assurance mechanism
- Defect information table
- Defect sector management of the allocation structure 0
- Defect sector management of the allocation structure 1

[0119]

[Effect of the Invention]

With the disc format of the present invention, an individual can record and reproduce compressed video and compressed audio signals simply in a home.

[BRIEF DESCRIPTION OF THE DRAWINGS]

[FIG. 1]

FIG. 1 is a view illustrating a structure of a volume information descriptor.

[FIG. 2]

FIG. 2 is a view illustrating a structure of logical volume information.

[FIG. 3]

FIG. 3 is a view illustrating a file system.

[FIG. 4]

FIG. 4 is a view illustrating a parent link, a next link and a child link.

[Description of Reference Symbols]

None

[NAME OF THE DOCUMENT] Abstract

[ABSTRACT]

[SUBJECT] To implement a file system for allowing an individual to record and reproduce AV signals simply in a home.

[SOLVING MEANS] In a file system for a recording and/or reproduction apparatus which uses a disc type recording medium, management information of an AV file system for recording AV data is recorded in an MIA, and the MIA is recorded at least at two locations of a logical volume.

[SELECTED FIGURE] None